บทที่ 5
การประยุกต์ใช้เจนเนติกอัลกอริทึมกับปัญหาการจัดผังโรงงาน

เนื้อหาในบทนี้กล่าวถึงการนำ GAs มาประยุกต์ใช้ในการแก้ปัญหาการจัดผังโรงงานของปัญหาการจัดผังโรงงาน ข้อจำกัดและสมมุติฐานของแบบจำลอง และขั้นตอนการเข้ารหัสสตริง การพิจารณาถึงค่าพารามิเตอร์และโอเปอร์เรเตอร์ต่างๆที่ใช้ในงานวิจัย และจะแสดงให้เห็นถึงวิธีการคำนวณปัญหาผังโรงงานตัวอย่างด้วย GAs โดยพิจารณาถึงเชิงคุณภาพอย่างละเอียด

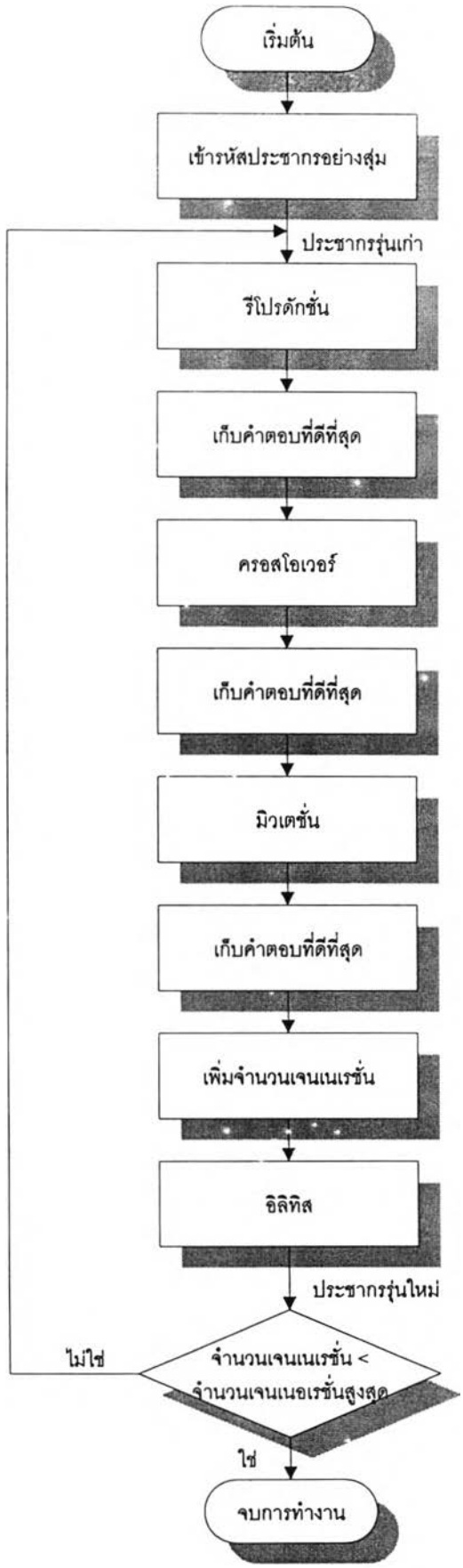
5.1 ข้อจำกัดและสมมติฐานของแบบจำลอง

ข้อจำกัดและสมมติฐานของแบบจำลองที่ใช้ในงานวิจัยมีดังต่อไปนี้

- ขนาดพื้นที่ของผังโรงงานของแต่ละแผนกมีขนาดพื้นที่ขนาดเท่ากันเป็น 1 ตารางหน่วย
- ลักษณะพื้นที่ของผังโรงงานทั้งหมดเป็นรูปสี่เหลี่ยม
- จำนวนแผนก $\leq$ จำนวนพื้นที่ ถ้าจำนวนแผนกน้อยกว่าจำนวนพื้นที่แผนกที่เหลือจะถูกกำหนดเป็นสถานีหุ่น (Dummy Station)
- แผนกแต่ละแผนกสามารถจัดวางบนพื้นที่ได้เพียง 1 ตำแหน่ง
- ระยะทางเป็นแบบยูคลิเดียน

5.2 ขั้นตอนของ GAs

สิ่งที่สำคัญของ GAs คือการเข้ารหัสสตริงให้ตรงกับรูปแบบปัญหาก่อน ขั้นตอนแรกเริ่มจากทำการสุ่มประชากรเริ่มต้นซึ่งอยู่ในรูปแบบสตริงจากนั้นจึงนำประชากรนั้นผ่านโอเปอร์เรชั่นของการรีโปรดักชั่น ครอสโอเวอร์ และ มิวเตชั่นตามลำดับ เพื่อให้ได้ประชากรใหม่ โดยที่ในแต่ละขั้นตอนมีการเก็บคำตอบที่ดีที่สุดไว้ จากนั้นประชากรคำตอบก็จะผ่านกระบวนการอิลิทิส (Elitist) จะเก็บสตริงคำตอบที่ดีที่สุดของเจนเนอเรชั่นนั่นไว้ ถ้าในเจนเนอเรชั่นต่อไปได้คำตอบที่ไม่ดีกว่าคำตอบของเจนเนเรชั่นปัจจุบัน ก็จะแทนที่คำตอบที่ไม่ดีนั้นด้วยคำตอบที่ดีของเจนเนอเรชั่นปัจจุบัน ขั้นตอนสุดท้ายเป็นการตรวจสอบสภาวะการหยุดทำงาน (Stopping Criteria) โดยตรวจสอบจากจำนวนเจนเนอเรชั่น ถ้าจำนวนเจนเนอเรชั่น มากกว่าหรือเท่ากับจำนวนเจนเนอเรชั่นสูงสุดให้หยุดทำงาน แต่ถ้าไม่ใช่ให้ทำการคำนวณใหม่อีกครั้งหนึ่ง ดังมีรายละเอียดดังรูปที่ 5.1



รูปที่ 5.1 ขั้นตอนของ GAs ที่ใช้ในงานวิจัย

5.3 การเข้ารหัสสตริง

ปัญหาตัวอย่างนี้เป็นการจัดผังโรงงานแบบใช้ข้อมูลเชิงคุณภาพ โดยที่แต่ละแผนกมีความสัมพันธ์ดังรูปที่ 4.21a การคำนวณหาระยะทางเพื่อคำนวณ TCR โดยรวมจำเป็นต้องอาศัยการเข้ารหัสสตริง วิธีการเข้ารหัสสตริงกับผังโรงงาน สมมติให้ตำแหน่งของผังโรงงานมีขนาด กว้าง*ยาว เป็น 2*5 หน่วยระยะทาง หรือ 10 แผนก (แต่ละแผนกมีขนาดกว้าง*ยาวเป็น 1*1 หน่วยระยะทาง) โดยที่ตำแหน่งพื้นที่เริ่มต้นจาก 0 จนถึง 9 แต่ละแผนกมีความสัมพันธ์ดังรูปที่ 4.21 ก ซึ่งมีวิธีการแสดงดังตัวอย่างต่อไปนี้

การเข้ารหัสสตริงของผังโรงงานขนาด 10 สถานี โดยกำหนดให้สตริงเริ่มต้นเป็น [3 4 7 6 1 0 8 9 5 2] ค่าแต่ละค่าในสตริงหรือยีนในโครโมโซมคือสถานีหนึ่งๆนั่นเอง สำหรับตำแหน่งในผังโรงงานคือตำแหน่งพื้นที่ดังแสดงในรูปที่ 5.2 รหัสของสตริงจะมีความสัมพันธ์กับตำแหน่งพื้นที่ของผังโรงงาน เช่น แผนกที่ 3 ในสตริงจะวางอยู่บนตำแหน่งที่ 0 และแผนกที่ 4 ในสตริงจะวางอยู่บนตำแหน่งที่ 4 เป็นต้น ส่วนแผนกที่เหลือวางเป็นลำดับต่อไปจากซ้ายไปขวาและบนลงล่างจนครบทั้งหมดดังรูปที่ 5.3 แสดงถึงการเข้ารหัสสตริงเริ่มต้น

0	1	2	3	4
5	6	7	8	9

รูปที่ 5.2 ตำแหน่งพื้นที่ของผังโรงงาน

3	4	7	6	1
0	8	9	5	2

รูปที่ 5.3 การจัดวางสถานีที่ 0 – 9 จากสตริงที่กำหนด

5.4 กระบวนการรีโปรดักชั่น

กระบวนการรีโปรดักชั่น คือกระบวนการในการคัดเลือกสตริงไปยังเจนเนอเรชั่นถัดไป โดยที่สตริงที่มีค่าความเหมาะสมสูงหรือสตริงที่ให้ค่าใช้จ่ายต่ำกว่ามีโอกาสที่จะถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปได้มากกว่าสตริงที่ให้ค่าใช้จ่ายสูงกว่า ซึ่งกระบวนการนี้เลียนแบบกระบวนการคัดเลือกทางธรรมชาติ ตามทฤษฎี Survival of Fittest ของ ชาร์ล ดาร์วิน

ดังนั้นในกระบวนการรีดักชั่นจะต้องสร้างฟิตเนสฟังก์ชั่นเพื่อเป็นการกำหนดว่าสตริงคำตอบของการจัดผังโรงงานที่สร้างขึ้นควรจะถูกคัดเลือกหรือไม่ โดยยกตัวอย่างการจัดผังโรงงานเชิงคุณภาพโดยพิจารณาถึงระยะทางระหว่างแผนก สตริงคำตอบที่มี TCR ที่น้อยกว่าควรมีโอกาสที่

จะถูกคัดเลือกได้มากกว่าสตริงคำตอบที่มี TCR ที่มากกว่า จากฟิตเนสฟังก์ชั่นทำให้สามารถกำหนดค่าความน่าจะเป็นที่สตริงใดๆจะถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปได้

สำหรับวิธีการคัดเลือกสตริงคำตอบไปยังเจนเนอเรชั่นถัดไป คือวิธีการของวงล้อรูลเล็ต (Roulette Wheel) (Goldberg, 1989) ซึ่งจะสุ่มจำนวนครั้งที่สตริงใดๆ จะถูกคัดเลือกจริงตามความน่าจะเป็นที่สร้างขึ้นจากฟิตเนสฟังก์ชั่น

ตารางที่ 5.1 กระบวนการรีโปรดักชั่นของปัญหาการจัดผังโรงงานด้วยข้อมูลเชิงคุณภาพ โดยพิจารณาระยะทางระหว่างแผนก

สตริงเริ่มต้น (1)	$f(x)=\sum\sum v(r_{ij})d_{ij}$ (2)	$\sum f(x)-f(x_i)$ (3)	$F_i=(\sum f(x)-f(x_i))/\sum f(x)$ (4)	ค่าเฉลี่ยของจำนวนครั้งที่ถูกคัดเลือก(5)	จำนวนครั้งที่ถูกคัดเลือกจริง(6)	สตริงใหม่ (7)	f(x) (8)
[6039274158]	101.9125	372.134	0.262	1.048	2	[6039274158]	101.9125
[6584109732]	123.782	350.265	0.246	.984	1	[6039274158]	101.9125
[7584236019]	124.176	349.871	0.246	.984	1	[6584109732]	123.782
[6574029831]	124.176	349.871	0.246	.984	0	[7584236019]	124.176
ผลรวม $\sum f(x)$	474.0465	1422.141	1.00	4	4		451.783
ค่าเฉลี่ย	118.512	355.535	.25	1	1		112.945

จากตารางที่ 5.1 แสดงถึงรีโปรดักชั่นของปัญหาการจัดผังโรงงานด้วยข้อมูลเชิงคุณภาพแบบวงล้อรูลเล็ท โดยกำหนดให้มีจำนวนประชากรหรือประชากรเริ่มต้นเป็น 4 สตริง

ให้คอลัมน์แรกเป็นสตริงที่สุ่มขึ้นซึ่งสามารถคำนวณ TCR ได้ตามฟิตเนสฟังก์ชั่นสมการที่ 4.5 ดังในคอลัมน์ที่ 2 สตริงที่มี TCR ที่น้อยที่สุดควรมีโอกาสที่จะอยู่รอดได้มากที่สุดจึงสร้างคอลัมน์ที่ 3 และ 4 เพื่อใช้ในการคำนวณขนาดของวงล้อรูลเล็ท ความน่าจะเป็นที่สตริงจะถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไป เนื่องจากฟิตเนสฟังก์ชั่นเป็นการหา TCR ที่ต่ำที่สุด ดังนั้นสตริงที่มี TCR ต่ำที่สุดควรจะมีโอกาสที่จะอยู่รอดต่อไปได้มากกว่าสตริงที่มี TCR สูงกว่า ดังจะเห็นได้ว่าสตริงตัวแรกมีความน่าจะเป็นในการอยู่รอด 26.2% ส่วนสตริงตัวสุดท้ายมีความน่าจะเป็นในการอยู่รอด 24.6% เราสามารถคำนวณหาจำนวนครั้งของสตริงที่จะถูกเลือกได้ดังคอลัมน์ที่ 5 คำนวณได้จาก F_i / $\overline{F_i}$ และคอลัมน์ที่ 6 เป็นจำนวนครั้งของโอกาสที่สตริงถูกคัดเลือกจริง จะเห็นได้ว่าสตริงตัวแรกถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปสองครั้ง สตริงตัวที่สองและสามถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปหนึ่งครั้ง และสตริงตัวสุดท้ายไม่ได้ถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปเลย เมื่อพิจารณาสตริงที่ผ่านการคัดเลือกแล้วในคอลัมน์สุดท้ายจะเห็นได้ว่า ผลรวมและค่าเฉลี่ยลดลงจากเดิมผลรวมของ TCR เป็น 474.0465 ลดลงเหลือ 451.783 และค่าเฉลี่ยลดลงจาก 118.512 เป็น 112.945

ตารางที่ 5.2 แสดงถึงกระบวนการรีโปรดักชั่นของปัญหาการจัดผังโรงงานด้วยข้อมูลเชิงคุณภาพโดยพิจารณาถึงความใกล้ชิดระหว่างแผนก

สตริงเริ่มต้น (1)	$f(x)=\sum\sum\delta_{ij}w_{ij}$ (2)	$F_i=f(x)/\sum f(x)$ (3)	ค่าเฉลี่ยของจำนวนครั้งที่ถูกคัดเลือก (4)	จำนวนครั้งที่ถูกคัดเลือกจริง (5)	สตริงใหม่ (6)	$f(x)$ (7)
[6850731942]	357	0.189	0.756	0	[6259710348]	389
[6259710348]	389	0.206	0.824	1	[2619385704]	541
[2619385704]	541	0.287	1.148	1	[2139065847]	601
[2139065847]	601	0.318	1.272	2	[2139065847]	601
ผลรวม $\sum f(x)$	1888	1	4	4		2132
ค่าเฉลี่ย	472	0.25	1	1		533

จากตารางที่ 5.2 แสดงถึงรีโปรดักชั่นของปัญหาการจัดผังโรงงานด้วยข้อมูลเชิงคุณภาพโดยพิจารณาถึงความใกล้ชิดระหว่างแผนกของสมการที่ 3.4 แบบวงล้อรูลเล็ท โดยกำหนดให้มีจำนวนประชากรเป็น 4 สตริง

สตริงตัวแรกมีค่าฟิตเนสที่น้อยที่สุดและสตริงตัวสุดท้ายมีค่าฟิตเนลมากที่สุด สตริงที่มี TCR ที่มากที่สุดควรมีโอกาสที่จะอยู่รอดได้มากที่สุด จะเห็นได้ว่าสตริงตัวแรกมีความน่าจะเป็นในการอยู่รอด 18.9% ส่วนสตริงตัวสุดท้ายมีความน่าจะเป็นในการอยู่รอด 31.8% จะเห็นได้ว่าสตริงสุดท้ายถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปสองครั้ง สตริงตัวที่สองและสามถูกคัดเลือกไปยังเจนเนเรชั่นต่อไปหนึ่งครั้ง และสตริงตัวแรกไม่ได้ถูกคัดเลือกไปยังเจนเนอเรชั่นต่อไปเลย เมื่อพิจารณาสตริงที่ผ่านการคัดเลือกแล้วในคอลัมน์สุดท้ายจะเห็นได้ว่า ผลรวมและค่าเฉลี่ยของ TCR มากขึ้นจากเดิมผลรวมของค่า TCR เป็น 472 เพิ่มขึ้นเป็น 533 และค่าเฉลี่ยของ TCR เพิ่มจากจาก 1888 เป็น 2132

กระบวนการคัดเลือกประชากรใหม่จะใช้หลักการของวงล้อรูลเล็ท ที่มีขนาดความกว้างของสล็อท (Slot) ตามค่าความเหมาะสม การสร้างวงล้อรูลเล็ทมีวิธีการดังต่อไปนี้

●กำหนดให้สตริงแต่ละตัวเป็น str_i ซึ่งสามารถคำนวณค่าความเหมาะสมของสตริงแต่ละตัว $eval(str_i)$ หรือ TCR ของสตริงเป็น c_i แต่ละตัว โดยที่ i = 1...pop (โดย pop คือจำนวนสตริงในแต่ละรุ่นประชากร)

●หาค่าความเหมาะสมโดยรวมของประชากร เนื่องจากสมการวัตถุประสงค์เป็นการหาค่าที่น้อยที่สุด ดังนั้นค่าที่น้อยที่สุดในประชากรจึงควรมีความน่าจะเป็นของการคัดเลือก (p_i) มากที่สุด ขนาดของช่องสล็อทในวงล้อรูลเล็ทสำหรับ TCR ที่มีค่าน้อยที่สุดควรมีขนาดกว้างมากที่สุด

$$CostSum = \sum_{i=1}^{pop} eval(str_i)$$

คำนวณหาค่า $CostSum\text{-}c_i$ และผลรวมของ $CostSum\text{-}c_i$ เป็น $Sum_of_(CostSum\text{-}c_i)$

●คำนวณหาค่าความน่าจะเป็นการคัดเลือก p_i ของสตริงแต่ละตัว

$$p_i = (CostSum - c_i)/Sum_of_(CostSum - c_i)$$

●คำนวณค่าความน่าจะเป็นสะสม (Cumulative Probability) ของสตริงแต่ละตัว (q_i) โดยที่ i = 1...pop

$$q_i = \sum_{j=1}^{i} p_i$$

กระบวนการคัดเลือกจะหมุนวงล้อรูลเล็ทเท่ากับจำนวน pop ครั้ง ในแต่ละครั้งจะได้สตริงสำหรับประชากรรุ่นต่อไป ดังมีรายละเอียดดังต่อไปนี้

●สร้างค่าสุ่ม (Random Number) r ที่มีค่าอยู่ในช่วง [0...1]

●ถ้า $r < q_i$ เลือกสตริงตัวแรก ถ้าไม่ใช่ เลือกสตริงตัวที่ i โดยที่ i อยู่ระหว่าง ($2 \leq i \leq pop$) หรือ ($q_{i-1} \leq r \leq q_i$)

ในบางครั้งสตริงที่ถูกคัดเลือกแล้วบางตัวบางตัวอาจถูกเลือกได้มากกว่า 1 ครั้ง

5.5 กระบวนการครอสโอเวอร์

การครอสโอเวอร์คือกระบวนการจับคู่สตริง แล้วแลกเปลี่ยนส่วนซึ่งกันและกันเพื่อให้เกิดสตริงใหม่ขึ้น

เนื่องจากปัญหาของการจัดผังโรงงานไม่ได้ใช้การเข้ารหัสแบบเลขฐานสอง การครอสโอเวอร์แบบธรรมดา หรือการครอสโอเวอร์แบบตำแหน่งเดียวไม่สามารถนำมาใช้ได้ เนื่องจากการครอสโอเวอร์แบบตำแหน่งเดียวอาจทำให้สตริงมีลักษณะที่ไม่ถูกต้อง กล่าวคืออาจทำให้ยีนบางตำแหน่งเกิดค่าซ้ำกัน ซึ่งทำให้ไม่สามารถคำนวณหาระยะทางของแผนกในผังโรงงานนั้นๆได้ Goldberg(1989) ได้เสนอวิธีการครอสโอเวอร์แบบสองจุด เพื่อแก้ปัญหาของการครอสโอเวอร์แบบธรรมดา การครอสโอเวอร์ที่นำเสนอมี 3 วิธีคือ การครอสโอเวอร์แบบ PMX (Partially Match Crossover) การครอสโอเวอร์แบบ OX (Order Crossover) และการครอสโอเวอร์แบบ CX (Cycle Crossover) ดังมีรายละเอียดดังต่อไปนี้

5.5.1 **การครอสโอเวอร์แบบ** PMX วิธีการของ PMX อย่างแรกคือการเลือกคู่สตริงพ่อแม่ขึ้นมาอย่างสุ่ม จากนั้นทำการเลือกตำแหน่งที่จะทำการครอสโอเวอร์อย่างสุ่มที่อยู่ในช่วง [1, l - 1] โดยที่ l คือความยาวของสตริง และยีนตัวแรกคือยีนหมายเลข 1 และยีนตัวสุดท้ายคือยีนหมายเลข l ขอบเขตของการครอสโอเวอร์อยู่ในช่วงเครื่องหมาย " | "

p_1 = [9 8 4 | 5 6 7 | 1 3 2 0]

p_2 = [8 7 1 | 2 3 0 | 9 5 4 6]

ในขั้นตอนต่อไปจะทำการสลับค่าระหว่างสตริงที่อยู่ในช่วง " | " นั่นคือตำแหน่งสุ่มอยู่ในช่วง [4, 6] ของโครโมโซมลูกหลานทั้งสอง โดยที่ค่าที่อยู่นอกเครื่องหมาย " | " และเป็นค่าที่ซ้ำกันกับค่าที่อยู่ในช่วงที่กำหนดให้เป็น x

จะเห็นได้ว่าตำแหน่งที่สองของสตริง p_2 มีค่าเป็น 7 ซึ่งซ้ำกับค่าที่อยู่ในช่วง " | " ของสตริง p_1 ก่อนทำการสลับ จึงเปลี่ยนค่าที่ซ้ำให้เป็น x ก่อนและตำแหน่งอื่นๆที่มีค่าซ้ำกันก็จะเปลี่ยนเป็น x ด้วยเช่นกัน

$$o_1 = [\,9\,8\,4\,|\,2\,3\,0\,|\,1\,x\,x\,x\,]$$

$$o_2 = [\,8\,x\,1\,|\,5\,6\,7\,|\,9\,5\,4\,x\,]$$

แล้วทำการแทนค่า (Map) ดังต่อไปนี้ 2 เป็น 5 และ 5 เป็น 2; 3 เป็น 6 และ 6 เป็น 3; 0 เป็น 7 และ 7 เป็น 0 การแทนค่าเหล่านี้ได้มาจากค่าที่อยู่ในช่วง " | " โดยพิจารณาตำแหน่งของสตริงที่ตรงกัน

สตริงที่ได้เมื่อทำการแทนที่แล้วคือ

$$o_1 = [\,9\,8\,4\,2\,3\,0\,1\,6\,5\,7\,]$$

$$o_2 = [\,8\,0\,1\,5\,6\,7\,9\,2\,4\,3\,]$$

5.5.2 **การครอสโอเวอร์แบบ OX** เช่นเดียวกับวิธีการของ PMX อย่างแรกคือการเลือกคู่สตริงพ่อแม่ขึ้นมาอย่างสุ่ม จากนั้นทำการเลือกตำแหน่งที่จะทำการครอสโอเวอร์อย่างสุ่ม ขอบเขตของการครอสโอเวอร์อยู่ในช่วงเครื่องหมาย " | "

$$p_1 = [\,8\,9\,4\,|\,5\,6\,7\,|\,1\,3\,2\,0\,]$$

$$p_2 = [\,8\,7\,1\,|\,2\,3\,0\,|\,9\,5\,4\,6\,]$$

สุ่มตำแหน่งที่จะทำการครอสโอเวอร์ได้อยู่ในช่วง [4, 6] จากนั้นเลือกคู่แม็ปที่อยู่ในช่วงการครอสโอเวอร์ถ้าค่าที่ตรงกันให้ทำเครื่องหมาย # เครื่องหมาย # หมายความว่าปล่อยตำแหน่งนั้นให้ว่าง สตริงลูกหลานจะเป็น

$$o_1 = [\,8\,9\,4\,|\,5\,6\,7\,|\,1\,\#\,\#\,\#\,]$$

$$o_2 = [\,8\,\#\,1\,|\,2\,3\,0\,|\,9\,\#\,4\,\#\,]$$

จากนั้นจะทำการเลื่อนตำแหน่ง โดยยึดถือตำแหน่งหลังช่วง " | " เป็นจุดอ้างอิง (สำหรับสตริง o_1 เป็น 1 และ สำหรับ o_2 เป็น 9) ได้สตริงลูกหลานดังต่อไปนี้

$$o_1 = [\,5\,6\,7\,|\,\#\,\#\,\#\,|\,1\,8\,9\,4\,]$$

$$o_2 = [\,2\,3\,0\,|\,\#\,\#\,\#\,|\,9\,4\,8\,1\,]$$

ขั้นตอนต่อไปจะเป็นการสลับตำแหน่งที่อยู่ในช่วงของการครอสโอเวอร์ของสตริงทั้งสองที่เหลือ สตริงที่ได้เมื่อทำการครอสโอเวอร์แล้วคือ

o_1 = [5 6 7 | 2 3 0 | 1 8 9 4]

o_2 = [2 3 0 | 5 6 7 | 9 4 8 1]

5.5.3 **การครอสโอเวอร์แบบ CX** ขั้นตอนแรกเลือกคู่สตริงพ่อแม่ขึ้นมาอย่างสุ่ม พิจารณาดังต่อไปนี้

p_1 = [2 0 3 4 8 5 9 1 7 6]

p_2 = [5 9 2 1 0 6 7 3 8 4]

พิจารณาที่ตำแหน่งแรกของสตริง p_1 และ p_2 คงเดิมไว้

o_1 = [2 - - - - - - - - -]

o_2 = [5 - - - - - - - - -]

ค่าตำแหน่งแรกของสตริง o_2 เป็น 5 นำมาพิจารณาค่าที่เป็น 5 (ตำแหน่งที่ 6) ของสตริง o_1 จะคงเดิมไว้ ส่วนค่าที่เป็น 6 (ตำแหน่งที่ 6) ของสตริง o_2 จะคงเดิมไว้เช่นกัน จะได้เป็น

o_1 = [2 - - - - 5 - - - -]

o_2 = [5 - - - - 6 - - - -]

จากค่าที่เป็น 6 ของสตริง o_2 นำไปพิจารณาต่อในสตริง o_1 ซึ่งจะคงเดิมไว้ ได้สตริงเป็น

o_1 = [2 - - - - 5 - - - 6]

o_2 = [5 - - - - 6 - - - 4]

ด้วยกระบวนการเดิมจะได้สตริงเป็น

o_1 = [2 - - 4 - 5 - - - 6]

o_2 = [5 - - 1 - 6 - - - 4]

↓

o_1 = [2 - - 4 - 5 - 1 - 6]

o_2 = [5 - - 1 - 6 - 3 - 4]

↓

o_1 = [2 - 3 4 - 5 - 1 - 6]

o_2 = [5 - 2 1 - 6 - 3 - 4]

กระบวนการในการเลือกตัวที่จะคงเดิมไว้จะสิ้นสุดต่อเมื่อ ค่าในตำแหน่งใดตำแหน่งหนึ่งของ o_2 มีค่าตรงกับค่าในตำแหน่งแรกของ สตริง o_1 จากนั้นจะทำการสลับค่าที่ยังคงเหลืออยู่ทั้งสองสตริงแบบ ตำแหน่งต่อตำแหน่ง ได้สตริงลูกหลานเป็น

o_1 = [2 9 3 4 0 5 7 1 8 6]

o_2 = [5 0 2 1 8 6 9 3 7 4]

ในขั้นสุดท้ายจะทำการคืนสตริงเข้าสู่เมทติ้งพูล

ในการครอสโอเวอร์ จะต้องมีการกำหนดพารามิเตอร์เพื่อหาจำนวนของสตริงจากประชากรทั้งหมดที่จะเกิดการครอสโอเวอร์ ค่าพารามิเตอร์นี้คือค่าความน่าจะเป็นของการครอสโอเวอร์ (Probability of Crossover, p_c) ซึ่งทำให้ได้จำนวนประชากรที่เกิดการครอสโอเวอร์คือ p_c*pop ขั้นตอนการสุ่มประชากรที่เกิดการครอสโอเวอร์ตามค่าความน่าจะเป็นของการ ครอสโอเวอร์ มีดังต่อไปนี้

- สร้างค่าสุ่ม (Random Number) r ที่มีค่าอยู่ในช่วง [0...1]
- ถ้า $r < p_c$ เลือกสตริงนั้นเพื่อทำการครอสโอเวอร์ตัวแรก
- จับคู่กับสตริงอีกหนึ่งตัวเพื่อทำการครอสโอเวอร์ ตามวิธีการของการครอสโอเวอร์แต่ละแบบ

5.6 กระบวนการมิวเตชั่น

กระบวนการมิวเตชั่น เป็นการเปลี่ยนค่าในบางตำแหน่งของสตริงเพื่อให้เกิดคำตอบใหม่ โดยเลือกตำแหน่งหรือยีนในโครโมโซมอย่างสุ่มและทำการสุ่มตำแหน่งอีกตำแหน่งในสตริงนั้นๆ เพื่อทำการสลับค่ากับตำแหน่งที่ถูกเลือกในครั้งแรก โดยมีการกำหนดค่าความน่าจะเป็นในการมิวเตชั่นเป็นจำนวนเปอร์เซ็นต์ เพื่อควบคุมจำนวนตำแหน่งหรือยีนที่จะมิวเตชั่น

สมมุติว่าสุ่มได้ยีนในตำแหน่งที่ 8 มีค่าเป็น 5 ดังสตริงในต่อไปนี้

[0 2 1 3 7 9 6 | 5 | 4 8]

จากนั้นทำการสุ่มเลือกตำแหน่งของยีนอีกตำแหน่งในสตริงเดียวกัน ได้ยีนในตำแหน่งที่ 2 มีค่าเป็น 2

[0 | 2 | 1 3 7 9 6 | 5 | 4 8]

เมื่อทำการมิวเตชั่นแล้วจะได้สตริงใหม่เป็น

[0 5 1 3 7 9 6 2 4 8]

จะเห็นได้ว่า กระบวนการของมิวเตชั่นเป็นการกระทำแบบตำแหน่งต่อตำแหน่ง จำเป็นต้องกำหนดความน่าจะเป็นของการมิวเตชั่น (Probability of Mutation) เพื่อกำหนดตำแหน่งที่จะทำการมิวเตชั่นชั่นหรือตำแหน่งที่คาดหวังว่าจะมีการมิวเตชั่นชั่น (Expected Mutation) หรือ p_m หาได้จาก $p_m.l.pop$ โดยที่ l คือความยาวของสตริงแต่ละตัวและ pop คือจำนวนประชากรหรือจำนวนสตริงทั้งหมด ดังนั้นทุกๆตำแหน่งในประชากรที่ผ่านการครอสโอเวอร์นั้นมีโอกาสที่จะถูกมิวเตชั่นชั่นได้ทุกตำแหน่ง

ขั้นตอนของการทำมิวเตชั่นมีดังต่อไปนี้

- สร้างค่าสุ่ม (Float) หรือ r อยู่ในช่วง [0....1] สำหรับตำแหน่งทุกตำแหน่ง
- ตำแหน่งใดที่ $r < p_m$ เลือกตำแหน่งนั้นเพื่อทำการมิวเตชั่น
- เลือกตำแหน่งอีกหนึ่งตำแหน่งในสตริงที่ตรงกับตำแหน่งที่ทำการเลือกครั้งแรก เพื่อทำการสลับค่ากับตำแหน่งแรกที่ถูกเลือก

หลังจากที่ผ่านกระบวนการคัดเลือก ครอสโอเวอร์ และมิวเตชั่นแล้ว ประชากรรุ่นใหม่ก็พร้อมที่จะนำไปสู่กระบวนการคัดเลือกต่อไป กระบวนการคัดเลือกของรุ่นประชากรรุ่นใหม่ใช้การกระบวนการของวงล้อรูเล็ทที่มีขนาดของช่องเป็นสัดส่วนกับค่าความเหมาะสม กระบวนการทั้งหมดจะเกิดขึ้นซ้ำๆกันซึ่งเป็นไปตามกระบวนการของ GAs

5.7 ตัวอย่างการหาคำตอบของเจนเนติกอัลกอริทึมกับปัญหาการจัดผังโรงงาน

เนื้อหาในส่วนนี้เป็นตัวอย่างการคำนวณการแก้ปัญหาการจัดผังโรงงานด้วย GAs โดยใช้ค่าการกระจายความน่าจะเป็น (Probability Distribution) แบบปกติ และวิธีการต่างๆดังที่ได้กล่าวมาแล้วคือ เริ่มจากสุ่มประชากรเริ่มต้น ผ่านกระบวนการรีโปรดักชั่น ครอสโอเวอร์และมิวเตชั่นจนครบจำนวนเจนเนอเรชั่นที่กำหนด เมื่อผ่านเวลาผ่านไปหลายรุ่นประชากรจะทำให้ได้คำตอบที่ดี (Optimum Solution) หรือใกล้เคียงคำตอบที่ดี

ตัวอย่างที่แสดงดังต่อไปนี้ เป็นการจำลอง GAs ที่มีขนาดประชากรเป็นสตริง 20 ตัว ค่าความน่าจะเป็นของการครอสโอเวอร์ p_c = 0.8 และค่าความน่าจะเป็นของการมิวเตชั่น p_m = 0.1 ขนาดผังโรงงานเป็น 10 สถานีซนิด 2 แถว 5 หลัก ค่าระดับความสัมพันธ์เป็นแบบลิเนียร์ โดยใช้แผนภูมิแสดงความสัมพันธ์จากรูปที่ 4.21 ก

ฟังก์ชั่นวัตถุประสงค์ที่ใช้ในแบบจำลองนี้คือฟังก์ชั่น

$$\text{Minimize } Z = \sum_{i=1}^{M} \sum_{j=i+1}^{M-1} V(r_{ij}) d_{ij}$$

โดยที่

d_{ij} คือระยะทางระหว่างแผนกแบบยูคลิเดียน

Z คือ TCR ที่พิจารณาถึงระยะทางระหว่างแผนก

$V(r_{ij})$ คือ คะแนนของความสัมพันธ์ระหว่างแผนก i และ j

โดยมีกระบวนการดังรูปที่ 3.2 ซึ่งมีรายละเอียดดังต่อไปนี้

5.7.1 สร้างประชากรเริ่มต้นอย่างสุ่ม

เริ่มต้นจากการสุ่มประชากรหรือสตริงเริ่มแรกที่เจนเนอเรชั่นแรกได้ดังต่อไปนี้

str_1 = 2 7 1 3 6 4 5 0 8 9

str_2 = 7 2 4 1 8 3 9 5 0 6

str_3 = 6 0 3 9 2 7 4 1 5 8

str_4 = 1 7 4 8 2 9 0 5 3 6

str_5 = 1 9 4 8 7 0 2 5 3 6

str_6 = 4 7 1 2 6 8 0 5 9 3

str_7 = 5 8 4 9 1 3 7 2 0 6

str_8 = 9 5 3 1 0 2 4 8 7 6

str_9 = 2 1 0 6 9 7 8 3 4 5

str_{10} = 1 8 3 4 7 9 0 5 2 6

str_{11} = 0 9 8 5 1 4 3 6 2 7

str_{12} = 8 5 3 2 1 6 7 0 4 9

str_{13} = 8 0 5 1 2 9 7 4 6 3

str_{14} = 8 7 5 6 1 3 4 2 9 0

str_{15} = 7 5 8 4 2 3 6 0 1 9

str_{16} = 0 1 7 3 4 5 8 2 9 6

str_{17} = 6 5 8 4 1 0 9 7 3 2

str_{18} = 6 5 7 4 0 2 9 8 3 1

str_{19} = 3 8 4 1 6 5 0 2 7 9

str_{20} = 4 1 8 5 6 3 9 2 0 7

ซึ่งสามารถคำนวณหา TCR ได้เป็น

$eval(str_1)$ = f(2 7 1 3 6 4 5 0 8 9) =121.953049

$eval(str_2)$ = f(7 2 4 1 8 3 9 5 0 6) =106.167961

$eval(str_3)$ = f(6 0 3 9 2 7 4 1 5 8) =101.912491

$eval(str_4)$ = f(1 7 4 8 2 9 0 5 3 6) =109.597557

$eval(str_5)$ = f(1 9 4 8 7 0 2 5 3 6) =119.302391

$eval(str_6)$ = f(4 7 1 2 6 8 0 5 9 3) =118.851532

$eval(str_7)$ = f(5 8 4 9 1 3 7 2 0 6) =103.146210

$eval(str_8)$ = f(9 5 3 1 0 2 4 8 7 6) =123.145554

$eval(str_9)$ = f(2 1 0 6 9 7 8 3 4 5) =113.788567

$eval(str_{10})$ = f(1 8 3 4 7 9 0 5 2 6) =105.143677

$eval(str_{11})$ = f(0 9 8 5 1 4 3 6 2 7) =113.468582

$eval(str_{12})$ = f(8 5 3 2 1 6 7 0 4 9) =113.993912

$eval(str_{13})$ = f(8 0 5 1 2 9 7 4 6 3) =107.922432

$eval(str_{14})$ = f(8 7 5 6 1 3 4 2 9 0) =115.865570

$eval(str_{15})$ = f(7 5 8 4 2 3 6 0 1 9) =124.175789

$eval(str_{16})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741

$eval(str_{17})$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990

$eval(str_{18})$ = f(6 5 7 4 0 2 9 8 3 1) =123.944954

$eval(str_{19})$ = f(3 8 4 1 6 5 0 2 7 9) =108.991356

$eval(str_{20})$ = f(4 1 8 5 6 3 9 2 0 7) =120.081703

Minimum is : 101.912491

Maximum is : 124.175789

Average is : 114.492273

Total is : 2289.845459

STD is : 7.320102

The Best Cost is = 101.912491 = [6 0 3 9 2 7 4 1 5 8]

ตัวอย่างสตริงตัวที่ 1 **แสดงถึงผังโรงงานดังรูปที่** 5.4

2	7	1	3	6
4	5	0	8	9

รูปที่ 5.4 ผังโรงงานคำตอบที่ได้จากสตริงตัวที่ 1

TCR **ของสตริง** str_1 **คำนวณได้ดังต่อไปนี้ (ระยะทางแบบยูคลิเดียน)**

Minimize $Z = \sum_{i=1}^{M} \sum_{j=i+1}^{M-1} V(r_{ij}) d_{ij}$

$$= V(r_{01})d_{01} + V(r_{02})d_{02} + V(r_{03})d_{03} + V(r_{04})d_{04} + V(r_{05})d_{05} + V(r_{06})d_{06} + V(r_{07})d_{07}$$
$$+ V(r_{08})d_{08} + V(r_{09})d_{09}$$
$$+ V(r_{12})d_{12} + V(r_{13})d_{13} + V(r_{14})d_{14} + V(r_{15})d_{15} + V(r_{16})d_{16} + V(r_{17})d_{17} + V(r_{18})d_{18}$$
$$+ V(r_{19})d_{19}$$
$$+ V(r_{23})d_{23} + V(r_{24})d_{24} + V(r_{25})d_{25} + V(r_{26})d_{26} + V(r_{27})d_{27} + V(r_{28})d_{28} + V(r_{29})d_{29}$$
$$+ V(r_{34})d_{34} + V(r_{35})d_{35} + V(r_{36})d_{36} + V(r_{37})d_{37} + V(r_{38})d_{38} + V(r_{39})d_{39}$$
$$+ V(r_{45})d_{45} + V(r_{46})d_{46} + V(r_{47})d_{47} + V(r_{48})d_{48} + V(r_{49})d_{49}$$
$$+ V(r_{56})d_{56} + V(r_{67})d_{67} + V(r_{58})d_{58} + V(r_{59})d_{59}$$
$$+ V(r_{67})d_{67} + V(r_{68})d_{68} + V(r_{69})d_{69}$$
$$+ V(r_{78})d_{78} + V(r_{79})d_{79}$$
$$+ V(r_{89})d_{89}$$

$$= 2*1 + 3*\sqrt{5} + 2*\sqrt{2} + 4*2 + 0*1 + 3*\sqrt{5} + (-1)*\sqrt{2} + 1*1 + 3*2$$
$$+ 0*2 + 2*1 + 0*\sqrt{5} + 1*\sqrt{2} + 2*2 + 1*1 + 1*\sqrt{2} + 2*\sqrt{5}$$
$$+ 2*3 + 3*1 + 0*\sqrt{2} + 2*4 + 0*1 + 2*\sqrt{10} + 0*\sqrt{17}$$
$$+ 1*\sqrt{10} + 1*\sqrt{5} + 0*1 + 0*2 + 0*1 + 4*\sqrt{2}$$
$$+ 0*1 + 4*\sqrt{17} + 2*\sqrt{2} + 0*3 + 0*4$$
$$+ 0*\sqrt{10} + 0*1 + 1*2 + 0*3$$
$$+ 1*3 + 0*\sqrt{2} + (-1*1)$$
$$+ 2*\sqrt{5} + 4*\sqrt{10}$$
$$+ 1*1$$
$$=121.953$$

สำหรับสตริงตัวอื่นๆก็คำนวณในลักษณะเช่นเดียวกัน เมื่อเริ่มต้นสุ่มค่าสตริงขึ้นมาชุดแรก จะเห็นได้ว่า ผังโรงงานที่มี TCR น้อยที่สุดในประชากรรุ่นนี้ได้แก่สตริง str_3 มี TCR เป็น 101.912491 ผังโรงงานทีมี TCR มากที่สุดในประชากรรุ่นนี้ได้แก่สตริง str_{15} มี TCR เป็น 124.175789 TCR โดยเฉลี่ยทั้งประชากรเป็น 114.492273 ส่วนเบี่ยงเบนมาตรฐาน (STD) เป็น 7.320102 ผลรวมของ TCR ทั้งหมดคือ 2289.845459

5.7.2 กระบวนการรีโปรดักชั่น

การทำรีโปรดักชั่นใช้วิธีการของวงล้อรูลเล็ทโดยที่ TCR ของสตริงแต่ละตัวเป็นอัตราส่วนกับขนาดของช่องในวงล้อรูลเล็ท สตริงที่มีค่าความเหมาะสมสูงจะถูกคัดเลือกสำหรับทำการรีโปรดักชั่น เนื่องจากปัญหานี้เป็นการหา TCR ที่น้อยที่สุดจึงจำเป็นต้องหาขนาดของช่องในวงล้อรูเล็ทโดยการนำ TCR ของสตริง c_i แต่ละตัวมาลบออกจากผลรวมของ TCR ทั้งหมดในประชากรทำให้ได้คำตอบเป็น *CostSum*- c_i จากนั้นจึงหาผลรวมของ *CostSum*- c_i ได้เป็น *Sum_of_*(*CostSum*-c_i) ขนาดของสล็อทในวงล้อรูเล็ทโดยการนำ *CostSum*- c_i ของสตริงแต่ละตัวหารด้วย *Sum_of_* (*CostSum*-c_i) โดยที่ขนาดของวงล้อรูเล็ทโดยรวมเป็น 100 เปอร์เซ็นต์

CostSum = 2289.845459

CostSum- c_i ของสตริงแต่ละตัวคือ

[1] 2289.845459 - 121.953049 = 2167.892334

[2] 2289.845459 - 106.167961 = 2187.153809

[3] 2289.845459 - 101.912491 = 2191.347656

[4] 2289.845459 - 109.597557 = 2183.645996

[5] 2289.845459 - 119.302391 = 2173.971191

[6] 2289.845459 - 118.851532 = 2174.459961

[7] 2289.845459 - 103.146210 = 2190.163574

[8] 2289.845459 - 123.145554 = 2170.102783

[9] 2289.845459 - 113.788567 = 2179.537842

[10] 2289.845459 - 105.143677 = 2188.146484

[11] 2289.845459 - 113.468582 = 2179.787598

[12] 2289.845459 - 113.993912 = 2179.294922

[13] 2289.845459 - 107.922432 = 2185.368408

[14] 2289.845459 - 115.865570 = 2177.401611

[15] 2289.845459 - 124.175789 = 2169.122314

[16] 2289.845459 - 114.609741 = 2178.720703

[17] 2289.845459 - 123.781990 = 2169.511230

[18] 2289.845459 - 123.944954 = 2169.384033

[19] 2289.845459 - 108.991356 = 2184.338135

[20] 2289.845459 - 120.081703 = 2173.189453

ผลรวมของ $CostSum\text{-}\ c_i$ คือ $Sum_of_(CostSum\text{-}\ c_i)$ = 43572.539063 ต่อไปเป็นการคำนวณหาค่าความน่าจะเป็น p_i ของแต่ละสตริง str_i

p_i = $(CostSum\text{-}\ c_i)_i / Sum_of_(CostSum\text{-}\ c_i)$

p_1 = 2167.892334 / 43572.539063 = 0.049754

p_2 = 2187.153809 / 43572.539063 = 0.050196

p_3 = 2191.347656 / 43572.539063 = 0.050292

p_4 = 2183.645996 / 43572.539063 = 0.050115

p_5 = 2173.971191 / 43572.539063 = 0.049893

p_6 = 2174.459961 / 43572.539063 = 0.049904

p_7 = 2190.163574 / 43572.539063 = 0.050265

p_8 = 2170.102783 / 43572.539063 = 0.049804

p_9 = 2179.537842 / 43572.539063 = 0.050021

p_{10} = 2188.146484 / 43572.539063 = 0.050218

p_{11} = 2179.787598 / 43572.539063 = 0.050027

p_{12} = 2179.294922 / 43572.539063 = 0.050015

p_{13} = 2185.368408 / 43572.539063 = 0.050155

p_{14} = 2177.401611 / 43572.539063 = 0.049972

p_{15} = 2169.122314 / 43572.539063 = 0.049782

p_{16} = 2178.720703 / 43572.539063 = 0.050002

p_{17} = 2169.511230 / 43572.539063 = 0.049791

p_{18} = 2169.384033 / 43572.539063 = 0.049788

p_{19} = 2184.338135 / 43572.539063 = 0.050131

p_{20} = 2173.189453 / 43572.539063 = 0.049875

จะเห็นได้ว่าสตริงตัวที่ดีที่สุดคือ str_3 มีขนาดเป็น 5.0292 % ของทั้งหมด ส่วนสตริงตัวที่แย่ที่สุดคือสตริงตัวที่ str_{15} มีขนาดเป็น 4.9782 % ของทั้งหมด สตริงตัวที่ดีที่สุดมีโอกาสถูกคัดเลือกได้มากกว่า

ต่อไปก็จะเป็นการหาค่าความน่าจะเป็นสะสม q_i สำหรับสตริงแต่ตัว

q_1 = 0.049754	q_2 = 0.099949	q_3 = 0.150241	q_4 = 0.200356	q_5 = 0.250250
q_6 = 0.300154	q_7 = 0.350419	q_8 = 0.400223	q_9 = 0.450244	q_{10} = 0.500462
q_{11} = 0.550489	q_{12} = 0.600504	q_{13} = 0.650659	q_{14} = 0.700631	q_{15} = 0.750413
q_{16} = 0.800415	q_{17} = 0.850206	q_{18} = 0.899994	q_{19} = 0.950125	q_{20} = 1.000000

จากนั้นจะทำการหมุนวงล้อรูเล็ท 20 ครั้ง การหมุนแต่ละครั้งหมายถึงการคัดเลือกสตริงเข้าไปสู่ประชากรใหม่ สมมุติค่าสุ่มจำนวน 20 ค่า ที่อยู่ในช่วง [0....1]

0.507520	0.548778	0.846160	0.815016	0.400760
0.886539	0.265524	0.336242	0.738740	0.814736
0.908088	0.778944	0.323392	0.754216	0.441124
0.688788	0.108368	0.664040	0.791954	0.410321

ค่าสุ่มตัวแรก r = 0.507520 มีค่ามากกว่า q_{10} แต่น้อยกว่า q_{11} เลือกเอาตัวที่มากกว่าดังนั้นจึงเลือก c_{11} ไปสู่ประชากรรุ่นใหม่ ค่าสุ่มเหล่านี้สามารถชี้ไปยังสตริงซึ่งสรุปได้ดังต่อไปนี้

r_1 = 11	r_2 = 11	R_3 = 17	r_4 = 17	r_5 = 9
r_6 = 18	r_7 = 6	r_8 = 7	r_9 = 15	r_{10} = 17
r_{11} = 19	r_{12} = 16	r_{13} = 7	r_{14} = 16	r_{15} = 9
R_{16} = 14	r_{17} = 3	R_{18} = 14	r_{19} = 16	r_{20} = 9

สตริงที่ถูกเลือกไปสู่ประชากรใหม่มากที่สุดคือสตริง c_9 สตริงในประชากรรุ่นเก่าและสตริงที่ผ่านกระบวนการรีโปรดักชั่นเป็นดังต่อไปนี้

สตริงก่อนการรีโปรดักชั่น	สตริงที่ผ่านการการรีโปรดักชั่นแล้ว
str_1 = [2 7 1 3 6 4 5 0 8 9]	str_{11} = [0 9 8 5 1 4 3 6 2 7]
str_2 = [7 2 4 1 8 3 9 5 0 6]	str_{11} = [0 9 8 5 1 4 3 6 2 7]
str_3 = [6 0 3 9 2 7 4 1 5 8]	str_{17} = [6 5 8 4 1 0 9 7 3 2]
str_4 = [1 7 4 8 2 9 0 5 3 6]	str_{17} = [6 5 8 4 1 0 9 7 3 2]
str_5 = [1 9 4 8 7 0 2 5 3 6]	str_9 = [2 1 0 6 9 7 8 3 4 5]
str_6 = [4 7 1 2 6 8 0 5 9 3]	str_{18} = [6 5 7 4 0 2 9 8 3 1]

str_7 = [5 8 4 9 1 3 7 2 0 6]
str_8 = [9 5 3 1 0 2 4 8 7 6]
str_9 = [2 1 0 6 9 7 8 3 4 5]
str_{10} = [1 8 3 4 7 9 0 5 2 6]
str_{11} = [0 9 8 5 1 4 3 6 2 7]
str_{12} = [8 5 3 2 1 6 7 0 4 9]
str_{13} = [8 0 5 1 2 9 7 4 6 3]
str_{14} = [8 7 5 6 1 3 4 2 9 0]
str_{15} = [7 5 8 4 2 3 6 0 1 9]
str_{16} = [0 1 7 3 4 5 8 2 9 6]
str_{17} = [6 5 8 4 1 0 9 7 3 2]
str_{18} = [6 5 7 4 0 2 9 8 3 1]
str_{19} = [3 8 4 1 6 5 0 2 7 9]
str_{20} = [4 1 8 5 6 3 9 2 0 7]

str_6 = [4 7 1 2 6 8 0 5 9 3]
str_7 = [5 8 4 9 1 3 7 2 0 6]
str_{15} = [7 5 8 4 2 3 6 0 1 9]
str_{17} = [6 5 8 4 1 0 9 7 3 2]
str_{19} = [3 8 4 1 6 5 0 2 7 9]
str_{16} = [0 1 7 3 4 5 8 2 9 6]
str_7 = [5 8 4 9 1 3 7 2 0 6]
str_{16} = [0 1 7 3 4 5 8 2 9 6]
str_9 = [2 1 0 6 9 7 8 3 4 5]
str_{14} = [8 7 5 6 1 3 4 2 9 0]
str_3 = [6 0 3 9 2 7 4 1 5 8]
str_{14} = [8 7 5 6 1 3 4 2 9 0]
str_{16} = [0 1 7 3 4 5 8 2 9 6]
str_9 = [2 1 0 6 9 7 8 3 4 5]

ทำการคืนค่าสตริงที่ผ่านการคัดเลือกแล้วเหล่านี้เข้าสู่เมทติ้งพูล

$eval(str_1)$ = f(0 9 8 5 1 4 3 6 2 7) =113.468582
$eval(str_2)$ = f(0 9 8 5 1 4 3 6 2 7) =113.468582
$eval(str_3)$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990
$eval(str_4)$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990
$eval(str_5)$ = f(2 1 0 6 9 7 8 3 4 5) =113.788567
$eval(str_6)$ = f(6 5 7 4 0 2 9 8 3 1) =123.944954
$eval(str_7)$ = f(4 7 1 2 6 8 0 5 9 3) =118.851532
$eval(str_8)$ = f(5 8 4 9 1 3 7 2 0 6) =103.146210
$eval(str_9)$ = f(7 5 8 4 2 3 6 0 1 9) =124.175789
$eval(str_{10})$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990
$eval(str_{11})$ = f(3 8 4 1 6 5 0 2 7 9) =108.991356
$eval(str_{12})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741
$eval(str_{13})$ = f(5 8 4 9 1 3 7 2 0 6) =103.146210
$eval(str_{14})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741

$eval(str_{15})$ = f(2 1 0 6 9 7 8 3 4 5) =113.788567

$eval(str_{16})$ = f(8 7 5 6 1 3 4 2 9 0) =115.865570

$eval(str_{17})$ = f(6 0 3 9 2 7 4 1 5 8) =101.912491

$eval(str_{18})$ = f(8 7 5 6 1 3 4 2 9 0) =115.865570

$eval(str_{19})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741

$eval(str_{20})$ = f(2 1 0 6 9 7 8 3 4 5) =113.788567

Minimum is : 101.912491

Maximum is : 124.175789

Average is : 114.968896

Total is : 2299.377930

STD is : 6.920040

The Best Cost is = 101.912491 = [6 0 3 9 2 7 4 1 5 8]

สตริง str_3 ถ่ายทอดไปยังลูกหลานได้เพียงครั้งเดียวเนื่องจากเป็นการสุ่มค่า จะเห็นได้ว่าค่าที่น้อยที่สุดยังคงเป็นค่าเดิมคือ 101.912491

5.7.3 กระบวนการครอสโอเวอร์

กระบวนการต่อไปเป็นการครอสโอเวอร์แบบ PMX ค่าความน่าจะเป็นของการครอสโอเวอร์ p_c = 0.8 อาจคาดได้ว่า 80% ของประชากรที่มีการครอสโอเวอร์ (โดยเฉลี่ย) นั่นคือ 0.8 * 20 = 16 กระบวนการของการครอสโอเวอร์เริ่มแรกคือทำการสุ่มค่า r อยู่ในช่วง [0....1] ได้ค่าดังต่อไปนี้

0.521180	0.383100	0.046380	0.129514	0.374916
0.342092	0.054210	0.773620	0.555335	0.472184
0.892500	0.433175	0.228612	0.945632	0.029774
0.563832	0.545851	0.970246	0.539160	0.575508

สตริงที่มีค่า r น้อยกว่า 0.8 มีอยู่ 17 ค่า ดังต่อไปนี้

str_1	str_2	str_3	str_4	str_5
str_6	str_7	str_8	str_9	str_{10}
str_{12}	str_{13}	str_{15}	str_{16}	str_{17}
str_{19}	str_{20}			

สตริงเหล่านี้จะถูกนำมาครอสโอเวอร์ การครอสโอเวอร์เป็นการหาค่าใหม่ที่อาศัยสตริงพ่อแม่สองตัว ในบางครั้งการสุ่มจำนวนสตริงพ่อแม่อาจไม่ได้ลงตัวพอดีเป็นเลขคู่ จึงอาจปัดลดลงมาหรือสุ่มเพิ่มสตริงอีกหนึ่งตัวเพื่อให้สามารถทำการครอสโอเวอร์ได้ ในที่นี้เลือกใช้การปัดลดลงมา จากนั้นจะเป็นการจับคู่สตริงพ่อแม่ดังต่อไปนี้

str_1	str_2
str_3	str_4
str_5	str_6
str_7	str_8
str_9	str_{10}
str_{12}	str_{13}
str_{15}	str_{16}
str_{17}	str_{19}

สตริงคู่แรกที่ทำการครอสโอเวอร์คือ str_1 และ str_2 จากนั้นทำการสุ่มตำแหน่งการครอสโอเวอร์สองตำแหน่งดังต่อไปนี้

คู่ str_1 และ str_2

ตำแหน่งแรกของการครอสโอเวอร์ = 4

ตำแหน่งที่สองของการครอสโอเวอร์ = 8

p_1 = [0 9 8 | 5 1 4 3 6 | 2 7]

p_2 = [0 9 8 | 5 1 4 3 6 | 2 7]

การครอสโอเวอร์แบบ PMX ตามที่ได้กล่าวมาแล้วข้างต้น

o_1 = [0 9 8 | 5 1 4 3 6 | 2 7]

o_2 = [0 9 8 | 5 1 4 3 6 | 2 7]

จะเห็นได้ว่าเมื่อสตริงพ่อแม่มีลักษณะเหมือนกัน เมื่อทำการครอสโอเวอร์แล้ว สตริงลูกหลานจะเหมือนกับสตริงพ่อแม่

คู่ str_3 และ str_4

ตำแหน่งแรกของการครอสโอเวอร์ = 6

ตำแหน่งที่สองของการครอสโอเวอร์ = 7

p_3 = [6 5 8 4 1 | 0 9 | 7 3 2]

p_4 = [6 5 8 4 1 | 0 9 | 7 3 2]

การครอสโอเวอร์แบบ PMX

o_3 = [6 5 8 4 1 | 0 9 | 7 3 2]

o_4 = [6 5 8 4 1 | 0 9 | 7 3 2]

คู่ str_5 และ str_6

ตำแหน่งแรกของการครอสโอเวอร์ = 7

ตำแหน่งที่สองของการครอสโอเวอร์ = 9

p_5 = [2 1 0 6 9 7 | 8 3 4 | 5]

p_6 = [6 5 7 4 0 2 | 9 8 3 | 1]

การครอสโอเวอร์แบบ PMX

o_5 = [2 1 0 6 4 7 | 9 8 3 | 5]

o_6 = [6 5 7 9 0 2 | 8 3 4 | 1]

คู่ str_7 และ str_8

ตำแหน่งแรกของการครอสโอเวอร์ = 4

ตำแหน่งที่สองของการครอสโอเวอร์ = 7

p_7 = [4 7 1 | 2 6 8 0 | 5 9 3]

p_8 = [5 8 4 | 9 1 3 7 | 2 0 6]

การครอสโอเวอร์แบบ PMX

o_7 = [4 0 6 | 9 1 3 7 | 5 2 8]

o_8 = [5 3 4 | 2 6 8 0 | 9 7 1]

คู่ str_9 และ str_{10}

ตำแหน่งแรกของการครอสโอเวอร์ = 3

ตำแหน่งที่สองของการครอสโอเวอร์ = 4

p_9 = [7 5 | 8 4 | 2 3 6 0 1 9]

p_{10} = [6 5 | 8 4 | 1 0 9 7 3 2]

การครอสโอเวอร์แบบ PMX

o_9 = [7 5 | 8 4 | 2 3 6 0 1 9]

o_{10} = [6 5 | 8 4 | 1 0 9 7 3 2]

คู่ str_{12} และ str_{13}

ตำแหน่งแรกของการครอสโอเวอร์ = 3

ตำแหน่งที่สองของการครอสโอเวอร์ = 9

p_{12} = [0 1 | 7 3 4 5 8 2 9 | 6]

p_{13} = [5 8 | 4 9 1 3 7 2 0 | 6]

การครอสโอเวอร์แบบ PMX

o_{12} = [5 8 | 4 9 1 3 7 2 0 | 6]

o_{13} = [0 1 | 7 3 4 5 8 2 9 | 6]

คู่ str_{15} และ str_{16}

ตำแหน่งแรกของการครอสโอเวอร์ = 5

ตำแหน่งที่สองของการครอสโอเวอร์ = 7

p_{15} = [2 1 0 6 | 9 7 8 | 3 4 5]

p_{16} = [8 7 5 6 | 1 3 4 | 2 9 0]

การครอสโอเวอร์แบบ PMX

o_{15} = [2 9 0 6 | 1 3 4 | 7 8 5]

o_{16} = [4 3 5 6 | 9 7 8 | 2 1 0]

คู่ str_{17} และ str_{19}

ตำแหน่งแรกของการครอสโอเวอร์ = 2

ตำแหน่งที่สองของการครอสโอเวอร์ = 8

p_{17} = [6 | 0 3 9 2 7 4 1 | 5 8]

p_{19} = [0 | 1 7 3 4 5 8 2 | 9 6]

การครอสโอเวอร์แบบ PMX

o_{17} = [6 | 1 7 3 4 5 8 2 | 9 0]

o_{19} = [8 | 0 3 9 2 7 4 1 | 5 6]

เมื่อทำการครอสโอเวอร์แล้วก็จะส่งสตริงเหล่านี้กลับไปยังเมทติ้งพูล สตริงที่ไม่ได้ถูกเลือกในการครอสโอเวอร์ก็จะคงอยู่เหมือนเดิม ประชากรใหม่เป็นดังต่อไปนี้

str_1 = 0 9 8 5 1 4 3 6 2 7

str_2 = 0 9 8 5 1 4 3 6 2 7

str_3 = 6 5 8 4 1 0 9 7 3 2

str_4 = 6 5 8 4 1 0 9 7 3 2

str_5 = 2 1 0 6 4 7 9 8 3 5

str_6 = 6 5 7 9 0 2 8 3 4 1

str_7 = 4 0 6 9 1 3 7 5 2 8

str_8 = 5 3 4 2 6 8 0 9 7 1

str_9 = 7 5 8 4 2 3 6 0 1 9

str_{10} = 6 5 8 4 1 0 9 7 3 2

str_{11} = 3 8 4 1 6 5 0 2 7 9

str_{12} = 5 8 4 9 1 3 7 2 0 6

str_{13} = 0 1 7 3 4 5 8 2 9 6

str_{14} = 0 1 7 3 4 5 8 2 9 6

str_{15} = 2 9 0 6 1 3 4 7 8 5

str_{16} = 4 3 5 6 9 7 8 2 1 0

str_{17} = 6 1 7 3 4 5 8 2 9 0

str_{18} = 8 7 5 6 1 3 4 2 9 0

str_{19} = 8 0 3 9 2 7 4 1 5 6

str_{20} = 2 1 0 6 9 7 8 3 4 5

สามารถคำนวนหา TCR ได้ดังต่อไปนี้

$eval(str_1)$ = f(0 9 8 5 1 4 3 6 2 7) =113.468582

$eval(str_2)$ = f(0 9 8 5 1 4 3 6 2 7) =113.468582

$eval(str_3)$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990

$eval(str_4)$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990

$eval(str_5)$ = f(2 1 0 6 4 7 9 8 3 5) =104.681168

$eval(str_6)$ = f(6 5 7 9 0 2 8 3 4 1) =113.746674

$eval(str_7)$ = f(4 0 6 9 1 3 7 5 2 8) =116.908974

$eval(str_8)$ = f(5 3 4 2 6 8 0 9 7 1) =99.287537

$eval(str_9)$ = f(7 5 8 4 2 3 6 0 1 9) =124.175789

$eval(str_{10})$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990

$eval(str_{11})$ = f(3 8 4 1 6 5 0 2 7 9) =108.991356

$eval(str_{12})$ = f(5 8 4 9 1 3 7 2 0 6) =103.146210

$eval(str_{13})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741

$eval(str_{14})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741

$eval(str_{15})$ = f(2 9 0 6 1 3 4 7 8 5) =97.915161

$eval(str_{16})$ = f(4 3 5 6 9 7 8 2 1 0) =120.370651

$eval(str_{17})$ = f(6 1 7 3 4 5 8 2 9 0) =103.666634

$eval(str_{18})$ = f(8 7 5 6 1 3 4 2 9 0) =115.865570

$eval(str_{19})$ = f(8 0 3 9 2 7 4 1 5 6) =114.677979

$eval(str_{20})$ = f(2 1 0 6 9 7 8 3 4 5) =113.788567

Minimum is : 97.915161

Maximum is : 124.175789

Average is : 113.236243

Total is : 2264.724854

STD is : 8.094949

The Best Cost is = 97.915161 = [2 9 0 6 1 3 4 7 8 5]

เมื่อผ่านการครอสโอเวอร์แล้วจะเห็นได้ว่าสตริงที่มีค่าที่ดีที่สุดลดลงจาก 101.912491 เป็น 97.915161 และค่าเฉลี่ยของทั้งประชากรลดลงจาก 114.968896 เป็น 113.236243

5.7.4 กระบวนการมิวเตชั่น

กระบวนการต่อไปเป็นคือมิวเตชั่น เป็นกระบวนการที่มีการกระทำแบบตำแหน่งต่อตำแหน่ง กำหนดความน่าจะเป็นของการมิวเตชั่น p_m = 0.10 นั้นคือจำนวนตำแหน่งที่จะผ่านการมิวเตชั่นเป็น 10 % ของจำนวนตำแหน่งทั้งหมด (โดยเฉลี่ย) ให้ l เป็นจำนวนตำแหน่งในสตริงแต่ละตัว ดังนั้นมีจำนวนตำแหน่งทั้งหมดในประชากรเป็น l * pop = 10 * 20 = 200 ตำแหน่ง อาจกล่าวได้ว่าจำนวนตำแหน่งทั้งหมดโดยเฉลี่ยที่จะผ่านการมิวเตชั่นคือ 200 * 0.1 = 20 ตำแหน่งต่อรุ่นประชากรในแต่ละรุ่น ทุกๆตำแหน่งมีโอกาสที่จะมิวเตชั่นเท่าๆกัน สร้างค่าสุ่ม r ที่อยู่ในช่วง [0....1] เท่ากับจำนวนตำแหน่งทั้งหมดที่อยู่ในประชากร ถ้า r < 0.1 ตำแหน่งนั้นจะถูกมิวเตชั่น

0.730588	0.507192	0.309280	0.534732	0.956341
0.552299	0.309268	0.350504	0.075810	0.081411
0.285840	0.809423	0.464448	0.827800	0.569285
0.305647	0.579825	0.673596	0.769579	0.299852
0.135083	0.755568	0.668804	0.685844	0.211848
0.040315	0.110584	0.434275	0.009345	0.732000
0.704138	0.947709	0.000368	0.670348	0.035088
0.309812	0.037070	0.223390	0.418956	0.741310
0.188356	0.591040	0.846700	0.631490	0.145348
0.209616	0.363040	0.186100	0.899364	0.855458
0.410248	0.643645	0.785136	0.555208	0.551456
0.038004	0.579024	0.664691	0.680982	0.099146
0.845620	0.325733	0.221159	0.022328	0.135432
0.200330	0.268578	0.077193	0.150059	0.974352
0.052672	0.558387	0.860016	0.405410	0.858250
0.950200	0.367730	0.338922	0.603076	0.214380
0.612016	0.578724	0.140794	0.192264	0.319912
0.278992	0.783000	0.601574	0.198160	0.785056
0.755786	0.752550	0.419516	0.990150	0.717294
0.625824	0.744073	0.337424	0.858246	0.428816
0.606820	0.137524	0.233628	0.268784	0.121050
0.769150	0.895072	0.560703	0.257344	0.872558
0.324316	0.335289	0.613467	0.647600	0.244756
0.783000	0.685920	0.090025	0.799140	0.191530
0.808394	0.737995	0.080778	0.148574	0.569255
0.831298	0.954784	0.149358	0.900822	0.931936
0.586072	0.522502	0.278976	0.722851	0.553052
0.596565	0.459527	0.177268	0.932200	0.899214
0.411066	0.713254	0.321534	0.301834	0.009532
0.325180	0.320640	0.239936	0.537690	0.195790
0.193320	0.299540	0.237715	0.349330	0.640510

0.766883	0.131968	0.565278	0.756250	0.137470
0.410595	0.523656	0.998522	0.054024	0.122036
0.561834	0.523242	0.350471	0.047540	0.335290
0.954784	0.078649	0.755744	0.047628	0.555374
0.141016	0.914872	0.539880	0.213104	0.369900
0.903360	0.443910	0.173164	0.113798	0.080100
0.655180	0.324875	0.431595	0.714300	0.580145
0.097210	0.659526	0.722748	0.653090	0.768002
0.186990	0.566176	0.669240	0.644317	0.574345

ค่า r ที่น้อยกว่า 0.1 มี 21 ตัว ดังตำแหน่งตำแหน่งดังต่อไปนี้

9	10	26	29	33
35	37	56	60	64
68	71	118	123	145
164	169	172	174	185
191				

การมิวเตชั่นของสตริงแตกต่างจากการมิวเตชั่นของสตริงที่เป็นเลขฐานสอง หลังจากที่ทราบว่าตำแหน่งใดที่จะทำการมิวเตชั่นแล้ว จำเป็นต้องตรวจสอบว่าตำแหน่งนั้นๆอยู่ในสตริงตัวใด เมื่อทราบว่าเป็นสตริงตัวใดแล้วจะทำการสุ่มเลือกตำแหน่งอีกหนึ่งตำแหน่งเพื่อทำการสลับค่าระหว่างตำแหน่งแรกกับตำแหน่งที่สองที่ผ่านการสุ่มมาแล้ว ดังต่อไปนี้

เริ่มต้นจากค่าสุ่มตำแหน่งแรก

Expected Mutation = 9 ค่า p_m = 0.075810

อยู่ในสตริง str_1 ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 9

[0 9 8 5 1 4 3 6 |2| 7]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 3

[0 9 |8| 5 1 4 3 6 |2| 7]

สลับค่าระหว่างตำแหน่งที่ 3 และ 9

[0 9 |2| 5 1 4 3 6 |8| 7]

Expected Mutation = 10 ค่า p_m = 0.081411

อยู่ในสตริง str_1 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 10

[0 9 2 5 1 4 3 6 8 |7|]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 7

[0 9 2 5 1 4 |3| 6 8 |7|]

สลับค่าระหว่างตำแหน่งที่ 7 **และ** 10

[0 9 2 5 1 4 |7| 6 8 |3|]

จะเห็นได้ว่าสตริงตัวแรกผ่านการมิวเตชั่นถึงสองครั้ง แต่ก็ไม่จำเป็นเสมอไป

Expected Mutation = 26 ค่า p_m = 0.040315

อยู่ในสตริง str_3 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 6

[6 5 8 4 1 |0| 9 7 3 2]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 9

[6 5 8 4 1 |0| 9 7 |3| 2]

สลับค่าระหว่างตำแหน่งที่ 6 **และ** 9

[6 5 8 4 1 |3| 9 7 |0| 2]

Expected Mutation = 29 ค่า p_m = 0.009345

อยู่ในสตริง str_3 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 9

[6 5 8 4 1 3 9 7 |0| 2]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 4

[6 5 8 |4| 1 3 9 7 |0| 2]

สลับค่าระหว่างตำแหน่งที่ 4 **และ** 9

[6 5 8 |0| 1 3 9 7 |4| 2]

Expected Mutation = 33 ค่า p_m = 0.000368

อยู่ในสตริง str_4 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 3

[6 5 |8| 4 1 0 9 7 3 2]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 4

[6 5 |8||4| 1 0 9 7 3 2]

สลับค่าระหว่างตำแหน่งที่ 3 **และ** 4

[6 5 |4||8| 1 0 9 7 3 2]

Expected Mutation = 35 ค่า p_m = 0.035088

อยู่ในสตริง str_4 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 5

[6 5 4 8 |1| 0 9 7 3 2]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 6

[6 5 4 8 |1||0| 9 7 3 2]

สลับค่าระหว่างตำแหน่งที่ 5 **และ** 6

[6 5 4 8 |0||1| 9 7 3 2]

Expected Mutation = 36 ค่า p_m = 0.037070

อยู่ในสตริง str_4 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 7

[6 5 4 8 0 1 |9| 7 3 2]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 8

[6 5 4 8 0 1 |9||7| 3 2]

สลับค่าระหว่างตำแหน่งที่ 7 **และ** 8

[6 5 4 8 0 1 |7||9| 3 2]

Expected Mutation = 56 ค่า p_m = 0.038004

อยู่ในสตริง str_6 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 6

[6 5 7 9 0 |2| 8 3 4 1]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 2

[6 |5| 7 9 0 |2| 8 3 4 1]

สลับค่าระหว่างตำแหน่งที่ 2 **และ** 6

[6 |2| 7 9 0 |5| 8 3 4 1]

Expected Mutation = 60 ค่า p_m = 0.099146

อยู่ในสตริง str_6 **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 10

[6 2 7 9 0 5 8 3 4 |1|]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 2

[6 |2| 7 9 0 5 8 3 4 |1|]

สลับค่าระหว่างตำแหน่งที่ 2 และ 9

[6 |1| 7 9 0 5 8 3 4 |2|]

Expected Mutation = 64 ค่า p_m = 0.022328

อยู่ในสตริง str_7 ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 4

[4 0 6 |9| 1 3 7 5 2 8]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 7

[4 0 6 |9| 1 3 |7| 5 2 8]

สลับค่าระหว่างตำแหน่งที่ 4 และ 7

[4 0 6 |7| 1 3 |9| 5 2 8]

Expected Mutation = 68 ค่า p_m = 0.077193

อยู่ในสตริง str_7 ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 8

[4 0 6 7 1 3 9 |5| 2 8]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 2

[4 |0| 6 7 1 3 9 |5| 2 8]

สลับค่าระหว่างตำแหน่งที่ 2 และ 8

[4 |5| 6 7 1 3 9 |0| 2 8]

Expected Mutation = 71 ค่า p_m = 0.052672

อยู่ในสตริง str_8 ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 1

[|5| 3 4 2 6 8 0 9 7 1]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 7

[|5| 3 4 2 6 8 |0| 9 7 1]

สลับค่าระหว่างตำแหน่งที่ 1 และ 7

[|0| 3 4 2 6 8 |5| 9 7 1]

Expected Mutation = 118 ค่า p_m = 0.090025

อยู่ในสตริง str_{12} ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 8

[5 8 4 9 1 3 7 |2| 0 6]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 10

[5 8 4 9 1 3 7 |2| 0 |6|]

สลับค่าระหว่างตำแหน่งที่ 8**และ** 10

[5 8 4 9 1 3 7 |6| 0 |2|]

Expected Mutation=123 ค่า p_m = 0.080778

อยู่ในสตริง str_{13} **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 3

[0 1 |7| 3 4 5 8 2 9 6]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 4

[0 1 |7||3| 4 5 8 2 9 6]

สลับค่าระหว่างตำแหน่งที่ 3 **และ** 4

[0 1 |3||7| 4 5 8 2 9 6]

Expected Mutation = 145 ค่า p_m = 0.009532

อยู่ในสตริง str_{15} **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 5

[2 9 0 6 |1| 3 4 7 8 5]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 1

[|2| 9 0 6 |1| 3 4 7 8 5]

สลับค่าระหว่างตำแหน่งที่ 1 **และ** 5

[|1| 9 0 6 |2| 3 4 7 8 5]

Expected Mutation = 164 ค่า p_m = 0.054024

อยู่ในสตริง str_{17} **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 4

[6 1 7 |3| 4 5 8 2 9 0]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 7

[6 1 7 |3| 4 5 |8| 2 9 0]

สลับค่าระหว่างตำแหน่งที่ 4 **และ** 7

[6 1 7 |8| 4 5 |3| 2 9 0]

Expected Mutation = 169 ค่า p_m = 0.047540

อยู่ในสตริง str_{17} **ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่** 9

[6 1 7 8 4 5 3 2 |9| 0]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 2

[6 |1| 7 8 4 5 3 2 |9| 0]

สลับค่าระหว่างตำแหน่งที่ 2 และ 9

[6 |9| 7 8 4 5 3 2 |1| 0]

Expected Mutation = 172 ค่า p_m = 0.078649

อยู่ในสตริง str_{18} ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 2

[8 |7| 5 6 1 3 4 2 9 0]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 6

[8 |7| 5 6 1 |3| 4 2 9 0]

สลับค่าระหว่างตำแหน่งที่ 2 และ 6

[8 |3| 5 6 1 |7| 4 2 9 0]

Expected Mutation = 174 ค่า p_m = 0.047628

อยู่ในสตริง str_{18} ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 4

[8 3 5 |6| 1 7 4 2 9 0]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 7

[8 3 5 |6| 1 7 4 |2| 9 0]

สลับค่าระหว่างตำแหน่งที่ 4 และ 7

[8 3 5 |2| 1 7 4 |6| 9 0]

Expected Mutation = 185 ค่า p_m = 0.080100

อยู่ในสตริง str_{19} ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 5

[8 0 3 9 |2| 7 4 1 5 6]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 10

[8 0 3 9 |2| 7 4 1 5 |6|]

สลับค่าระหว่างตำแหน่งที่ 5 และ 10

[8 0 3 9 |6| 7 4 1 5 |2|]

Expected Mutation = 191 ค่า p_m = 0.097210

อยู่ในสตริง str_{20} ตำแหน่งแรกที่ได้จากการสุ่มเป็นตำแหน่งที่ 1

[|2| 1 0 6 9 7 8 3 4 5]

ทำการสุ่มเลือกอีกตำแหน่งเป็นตำแหน่งที่ 4

[|2| 1 0 |6| 9 7 8 3 4 5]

สลับค่าระหว่างตำแหน่งที่ 1 และ 4

[|6| 1 0 |2| 9 7 8 3 4 5]

สตริงในประชากรใหม่ หลังจากผ่านการมิวเตชั่นแล้วคือ

str_1 = 0 9 2 5 1 4 7 6 8 3

str_2 = 0 9 8 5 1 4 3 6 2 7

str_3 = 6 5 8 0 1 3 9 7 4 2

str_4 = 6 5 4 8 0 1 7 9 3 2

str_5 = 2 1 0 6 4 7 9 8 3 5

str_6 = 6 1 7 9 0 5 8 3 4 2

str_7 = 4 5 6 7 1 3 9 0 2 8

str_8 = 0 3 4 2 6 8 5 9 7 1

str_9 = 2 9 0 6 1 3 4 7 8 5

str_{10} = 6 5 8 4 1 0 9 7 3 2

str_{11} = 3 8 4 1 6 5 0 2 7 9

str_{12} = 5 8 4 9 1 3 7 6 0 2

str_{13} = 0 1 3 7 4 5 8 2 9 6

str_{14} = 0 1 7 3 4 5 8 2 9 6

str_{15} = 1 9 0 6 2 3 4 7 8 5

str_{16} = 4 3 5 6 9 7 8 2 1 0

str_{17} = 6 9 7 8 4 5 3 2 1 0

str_{18} = 8 3 5 2 1 7 4 6 9 0

str_{19} = 8 0 3 9 6 7 4 1 5 2

str_{20} = 6 1 0 2 9 7 8 3 4 5

$eval(str_1) = f(0\,9\,2\,5\,1\,4\,7\,6\,8\,3) = 110.645058$

$eval(str_2) = f(0\,9\,8\,5\,1\,4\,3\,6\,2\,7) = 113.468582$

$eval(str_3) = f(6\,5\,8\,0\,1\,3\,9\,7\,4\,2) = 117.685760$

$eval(str_4)$ = f(6 5 4 8 0 1 7 9 3 2) =105.523109

$eval(str_5)$ = f(2 1 0 6 4 7 9 8 3 5) =104.681168

$eval(str_6)$ = f(6 1 7 9 0 5 8 3 4 2) =101.979012

$eval(str_7)$ = f(4 5 6 7 1 3 9 0 2 8) =108.823204

$eval(str_8)$ = f(0 3 4 2 6 8 5 9 7 1) =110.104675

$eval(str_9)$ = f(2 9 0 6 1 3 4 7 8 5) =97.915161

$eval(str_{10})$ = f(6 5 8 4 1 0 9 7 3 2) =123.781990

$eval(str_{11})$ = f(3 8 4 1 6 5 0 2 7 9) =108.991356

$eval(str_{12})$ = f(5 8 4 9 1 3 7 6 0 2) =109.878403

$eval(str_{13})$ = f(0 1 3 7 4 5 8 2 9 6) =108.676956

$eval(str_{14})$ = f(0 1 7 3 4 5 8 2 9 6) =114.609741

$eval(str_{15})$ = f(1 9 0 6 2 3 4 7 8 5) =100.534393

$eval(str_{16})$ = f(4 3 5 6 9 7 8 2 1 0) =120.370651

$eval(str_{17})$ = f(6 9 7 8 4 5 3 2 1 0) =115.578529

$eval(str_{18})$ = f(8 3 5 2 1 7 4 6 9 0) =113.757156

$eval(str_{19})$ = f(8 0 3 9 6 7 4 1 5 2) =116.568062

$eval(str_{20})$ = f(6 1 0 2 9 7 8 3 4 5) =108.788567

Minimum is : 97.915161

Maximum is : 123.781990

Average is : 110.618079

Total is : 2212.361572

STD is : 6.589276

The Best Cost is = 97.915161 = [2 9 0 6 1 3 4 7 8 5]

จากกระบวนการมิวเตชั่นที่ผ่านมายังไม่พบสตริงที่มีค่าน้อยกว่าสตริงก่อนกระบวนการมิวเตชั่น สตริงที่มีค่าน้อยที่สุดยังคงเป็น 97.915161 แต่ค่าเฉลี่ยของทั้งประชากรลดลงจาก 113.236243 เป็น 110.618079

เริ่มเจนเนอเรชั่นใหม่จากประชากรรุ่นที่ผ่านมาและทำกระบวนการต่างๆซ้ำเช่นเดิมจนเจนเนอเรชั่นมากกว่าหรือเท่ากับเจนเนอเรชั่นสูงสุดที่กำหนดไว้จึงเป็นการหยุดการคำนวณ

กระบวนการเหล่านี้จะเกิดขึ้นหลายครั้งโดยคาดหวังว่าจะได้คำตอบที่ดีขึ้นเรื่อยๆ คำตอบที่ดีที่สุดในแต่ละรุ่นประชากรจะถูกเก็บไว้แยกออกจากกัน แต่อย่างไรก็ดีจากแบบจำลองที่ผ่านมา

คำตอบที่ดีที่สุดในแต่ละรุ่นประชากรจะถูกไว้ที่ The Best Cost ถ้าผ่านกระบวนการต่างๆแล้วทำให้คำตอบที่ดีที่สุดแย่กว่าคำตอบในกระบวนการก่อนหน้านี้ ก็จะนำคำตอบในกระบวนการก่อนหน้านี้มาแทน เนื่องจากต้องการเก็บคำตอบที่ดีที่สุดไว้ ซึ่งวิธีนี้เป็นการบังคับให้เก็บคำตอบที่ดีที่สุดไว้ (Elitist)

5.8 สรุป

เนื่องจาก GAs เป็นการค้นหาคำตอบที่มีประสิทธิภาพสูง แต่การนำ GAs ไปใช้ในการแก้ปัญหาต่างๆนั้นจำเป็นต้องมีกระบวนการต่างๆในรายละเอียดปลีกย่อยเพื่อที่จะทำให้ GAs สามารถทำงานได้อย่างที่ต้องการ โดยเริ่มจาก การเข้ารหัสสตริงของปัญหาการจัดผังโรงงานทำได้โดยการกำหนดสตริงที่มีจำนวนยีนเท่ากับจำนวนสถานีขึ้นมาก่อน จากนั้นกำหนดขนาดของผังโรงงาน (แต่ละแผนกมีขนาด กว้าง*ยาวเป็น 1*1 หน่วยระยะทาง) แล้ววางแผนกแรกจากสตริงลงบนตำแหน่งพื้นที่ของผังโรงงานตำแหน่งแรกส่วนแผนกที่เหลือวางเป็นลำดับต่อไปจากซ้ายไปขวาและบนลงล่างจนครบทั้งหมด

การคัดเลือก (รีโปรดักชั่น) แบบวงล้อรูลเล็ทสามารถทำได้โดย

- สร้างค่าสุ่ม (random number) r ที่มีค่าอยู่ในช่วง [0...1]
- ถ้า $r < q_i$ เลือกสตริงตัวแรก ถ้าไม่ใช่ เลือกสตริงตัวที่ i โดยที่ i อยู่ระหว่าง ($2 \leq i \leq pop$) หรือ ($q_{i-1} \leq r \leq q_i$)

การครอสโอเวอร์ที่ใช้ในงานวิจัยมี 3 อย่างคือ การครอสโอเวอร์แบบ PMX (Partially Match Crossover) การครอสโอเวอร์แบบ OX (Order Crossover) และการครอสโอเวอร์แบบ CX (Cycle Crossover) โดยสามารถทำได้ดังต่อไปนี้

- สร้างค่าสุ่ม (Random Number) r ที่มีค่าอยู่ในช่วง [0...1]
- ถ้า $r < p_c$ เลือกสตริงนั้นเพื่อทำการครอสโอเวอร์ตัวแรก
- จับคู่กับสตริงอีกหนึ่งตัวเพื่อทำการครอสโอเวอร์ ตามขั้นตอนของการครอสโอเวอร์แต่ละแบบ

วิธีการมิวเตชั่นมีดังต่อไปนี้ สามารถทำได้ดังต่อไปนี้

- สร้างค่าสุ่ม (Float) หรือ r อยู่ในช่วง [0....1] สำหรับตำแหน่งทุกตำแหน่ง
- ตำแหน่งใดที่ $r < p_m$ เลือกตำแหน่งนั้นเพื่อทำการมิวเตชั่น
- เลือกตำแหน่งอีกหนึ่งตำแหน่งในสตริงที่ตรงกับตำแหน่งที่ทำการเลือกครั้งแรก เพื่อทำการสลับค่ากับตำแหน่งแรกที่ถูกเลือก